section 2

# Rapport du projet III :

# Gestion de Stock d’un Supermarché



## Sommaire

[Introduction](#_b2aiwqvshgzn)

[a. Contexte](#_ao26ab34mj06)

[b. Objectifs](#_oxuwuijnuaph)

[Méthodologie](#_vy65068h3u1)

[1. Choix des entités et définition des associations](#_qf1wbqyteoii)

[2. création de la base de données et des tables](#_gk760hy2ibts)

[3. Ajout des données](#_tlyy4h1y6nuc)

[4. Création des Opérations](#_ri79nc5692e4)

[5. Mise en place des vues et triggers](#_s05rza2wrsau)

[Résultats](#_gv7cjy9tom0x)

[1. Modèle Logique de données](#_vaxwtpgkgguf)

[2. création de la base de données et des tables](#_6bk9eqwdqvez)

[3. Ajout des données](#_aasc77pfztv9)

[4. création des requêtes](#_pf87ynk5bwm9)

[5. Définition des fonction et procédures de notre système](#_ietrqzriu7qz)

[6. mise en place des vues et triggers](#_apm9u3td5ckv)

[Conclusion](#_uardjybwaowm)

## Introduction

### Contexte

Dans un environnement commercial en constante évolution, la gestion efficace des stocks est cruciale pour assurer la disponibilité des produits, minimiser les coûts et maximiser la satisfaction des clients. Ce projet vise à développer un système de gestion des stocks pour un supermarché, permettant de suivre les produits, les commandes vers les fournisseurs, les fournisseurs eux-mêmes, ainsi que les niveaux de stock.

Le Système proposé répondra à plusieurs besoins critiques du supermarché:

* **Suivi des produits**: Enregistrer et gérer les informations sur chaque produit, y compris les description, les prix, et les quantités disponibles.
* **Gestion des commandes vers les fournisseurs:** Suivre les commandes passées aux fournisseurs, gérer les délais de livraison, et s’assurer que les stocks sont réapprovisionnés en temps voulu.
* **Gestion des fournisseurs:** Maintenir une base de données des fournisseurs, incluant leurs informations de contact, les conditions commerciales, et l’historique des transactions.
* **Suivi des stocks:** Surveiller les niveaux de stock en temps réel, générer des alertes pour les réapprovisionnements, et produire des rapports détaillés pour la gestion et la prise de décision.

### Objectifs

L’objectif principal de ce projet est de créer un système automatisé qui améliorera l’efficacité opérationnelle, réduira les erreurs humaines et permettra une gestion proactive des stocks. En outre, le système offrira des fonctionnalités de reporting avancées pour aider la direction à prendre des décisions informées et stratégiques.

Le présent rapport décrit en détail le processus de développement du système, depuis l’analyse des besoins jusqu’à la mise en œuvre, en passant par la conception et l'intégration des différentes composantes.

## 

## Méthodologie

Pour parvenir à un système qui répond pleinement aux exigences susmentionnées, nous avons choisi de procéder méthodiquement

### Choix des entités et définition des associations

Suivant les besoins critique du super marché, nous avons réussi a répertorié 04 entités et trois associations lors de notre analyse. Il s’agit de:

Entités

* **produit**: id\_produit, nom, description, prix, stock, id\_fournisseur, id\_categorie
* **Fournisseurs**: id\_fournisseur, nom\_fournisseur, contact, adresse, telephone
* **Commandes**: id\_commande, id\_produit, quantite, date\_commande
* **Categories**: id\_categorie, nom\_categorie

Associations

| Association | Cardinalité | Explication |
| --- | --- | --- |
| Contenir | (1.1, 0.N) | Une commande ne peut contenir qu’un seul produit ; et un produit peut être contenu dans zéro ou plusieur produit. |
| Appartenir | (1.1, 1.N) | Un produit appartient a une seule catégorie ; et une catégorie peut avoir plusieurs produits. |
| livrer | (1.1, 1.N) | un produit est livré par un seul fournisseur ; et un fournisseur peut livrer plusieurs produits. |

### création de la base de données et des tables

Suivant le rendu de notre MCD présenté dans la documentation technique, nous avons créé une base de donnée “gestion-stockDB” ; des tables à base du langage SQL et nous leur avons attribué les caractéristiques suivantes.

1. Commandes

| Caractéristique | Nom du champ | type | NULL? |
| --- | --- | --- | --- |
| cle de référence de la table | id\_commande | int primary key | non |
| indique la quantité du produit commandé | quantite | int | non |
| indique la date auquel la commande a été passée | date\_commande | date | non |

1. Produits

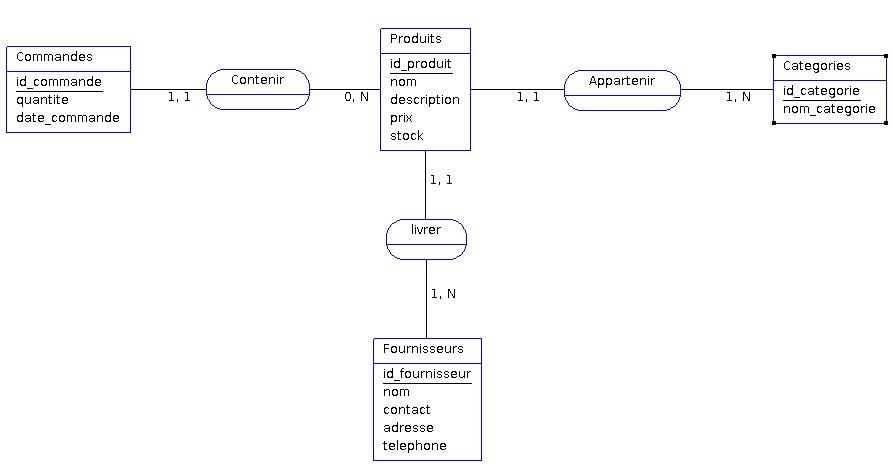
| Caractéristique | Nom du champ | type | NULL? |
| --- | --- | --- | --- |
| cle de référence de la table | id\_produit | int primary key | non |
| indique la quantité du produit commandé | nom\_produit | varchar | non |
| indique la description du produit | description | varchar | non |
| indique le prix du produit | prix | real | non |
| indique la quantité du produit restant en stock | stock | int | non |

1. Fournisseurs

| Caractéristique | Nom du champ | type | NULL? |
| --- | --- | --- | --- |
| cle de référence de la table | id\_fournisseur | int primary key | non |
| indique le nom du fournisseur | nom\_fournisseur | varchar | non |
| indique l’adresse email du fournisseur | contact | varchar | non |
| indique l’adresse du fournisseur | adresse | varchar | non |
| indique le numéro du fournisseur | telephone | varchar | non |

1. Categories

| Caractéristique | Nom du champ | type | NULL? |
| --- | --- | --- | --- |
| cle de référence de la table | id\_categorie | int primary key | non |
| indique le nom du fournisseur | nom\_categorie | varchar | non |



### Ajout des données

La création de nos différentes tables étant faite, nous avons pour une opérationnalisation de notre base de données inséré une série de données pour tester son bon fonctionnement.

Chacun des ces donnés remplit pleinement les caractéristiques que nous avons définies dans les tables ci-dessus. Nous avons respectivement dans notre base de données à l’heure actuelle.

* 03 Categories
* 03 Fournisseurs
* 03 Produits

### 

### Création des Opérations

Notre système en plus de stocker et gérer les produits et les fournisseurs fournit aussi une panoplie de requêtes, fonctions, procédures et autres pour faciliter les tâches de recherche de mise à jour et suppression de certaines données ayant une caractéristique bien précise.

Nous avons prévu des requête pour :

* Lister tous les produits d’une certaine catégorie
* trouver les fournisseurs d’un certain produit.
* lister les commandes en cours pour un produit
* compter le nombre total de commandes par produit

Nous avons des fonctions et procédure tel que :

* Une procédure pour ajouter une nouvelle commande
* Une fonction pour vérifier la disponibilité d’un produit

### 

### Mise en place des vues et triggers

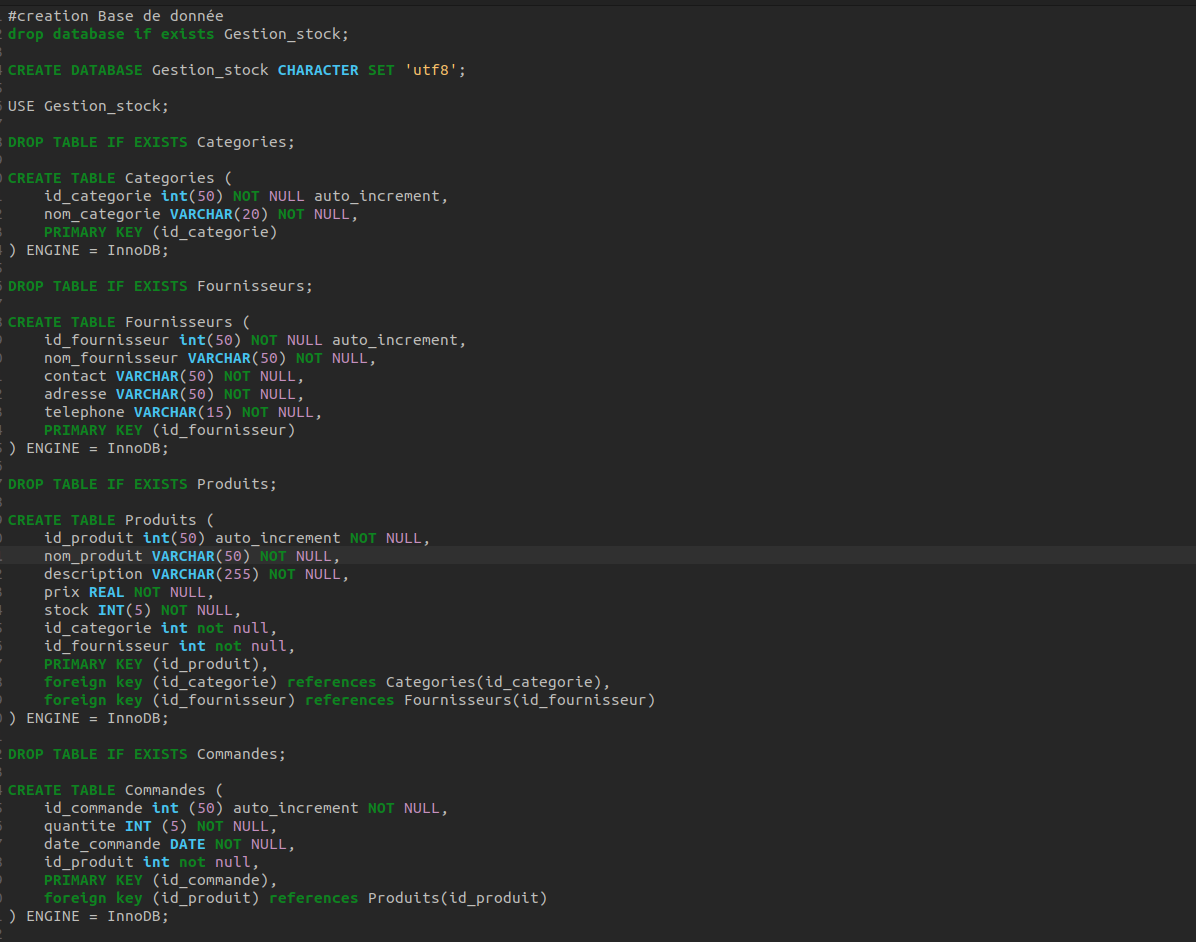
Les vues sont des objets de la base de données, constitués d’un nom, et d’une requête de sélection. Une fois qu’une vue est définie, on peut l’utiliser comme on le ferait avec une table ; table qui serait constituée par la requête définissant la vue. Pour notre projet nous avons défini une vue pour afficher les détails d’une commande (nom du produit, nom du fournisseur, quantité, date de commande).

Les triggers (ou déclencheurs) sont des objets de la base de données. Attachés à une table, ils vont déclencher l'exécution d’une instruction, ou d’un bloc d'instruction, lorsqu’une, ou plusieurs lignes sont insérées, supprimées ou modifiées dans la table à laquelle ils sont attachés. Pour notre projet nous avons créé un trigger pour mettre à jour le stock d’un produit à l'ajout et à la suppression d'une commande.

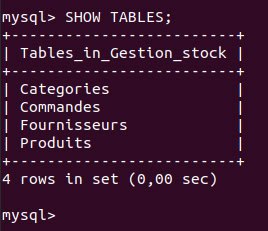
## Résultats

Dans cette section vous trouverez pour chaque étape de notre projet des captures d'écran montrant l’ensemble de nos actions sous forme de scripts SQL et le rendu lors de l'implémentation dans MySQL.

### création de la base de données et des tables



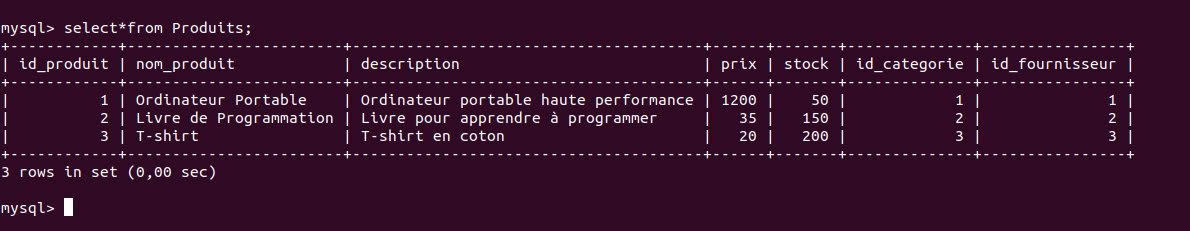
* affichage des tables



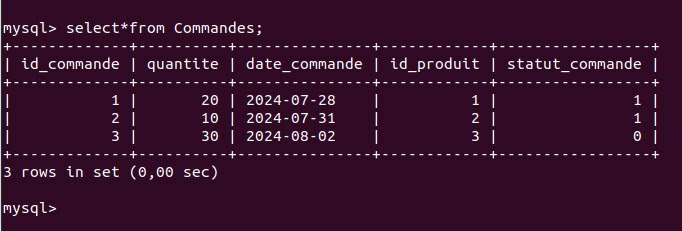
### Ajout des données

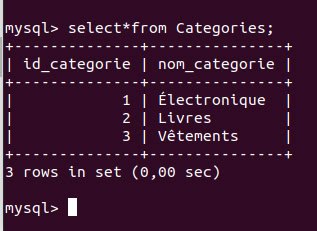


* Affichage des Tableaux avec les données

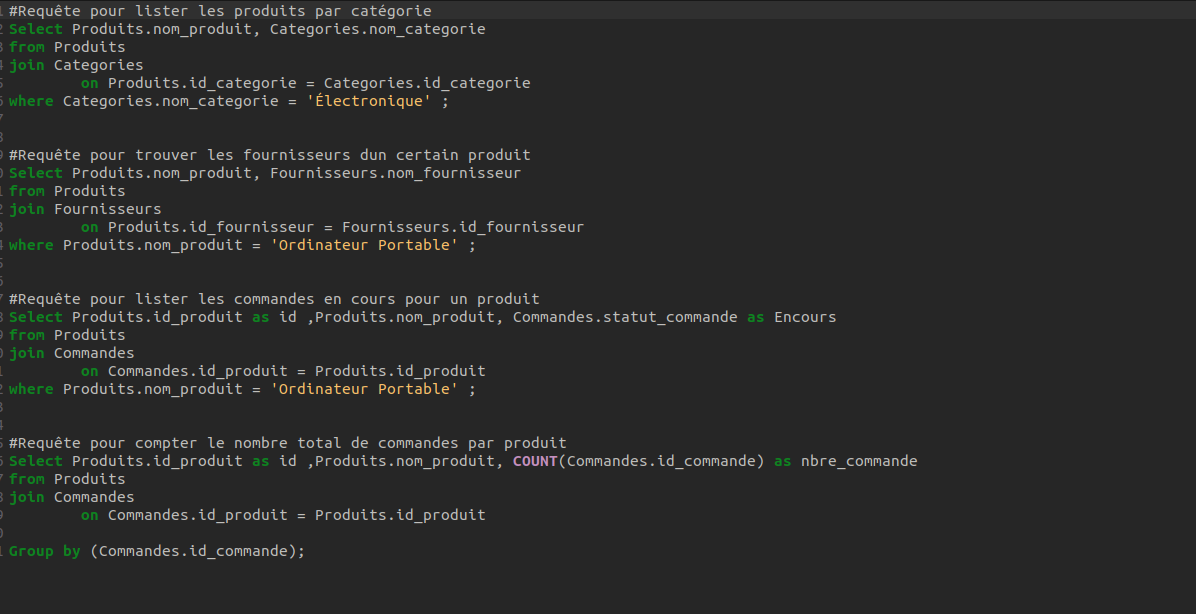




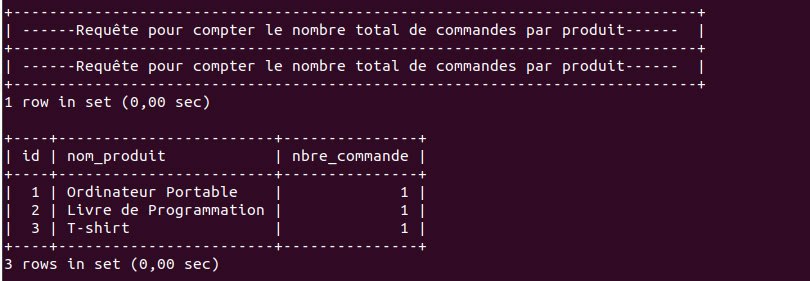


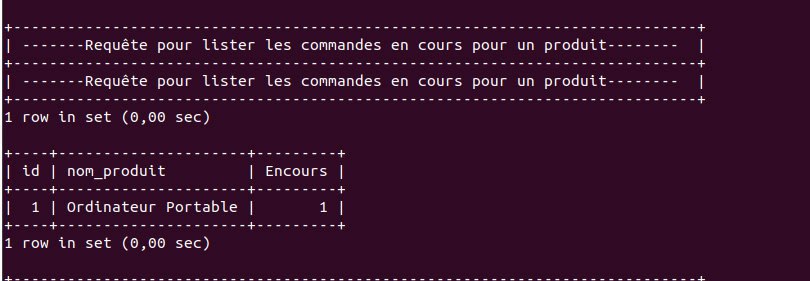


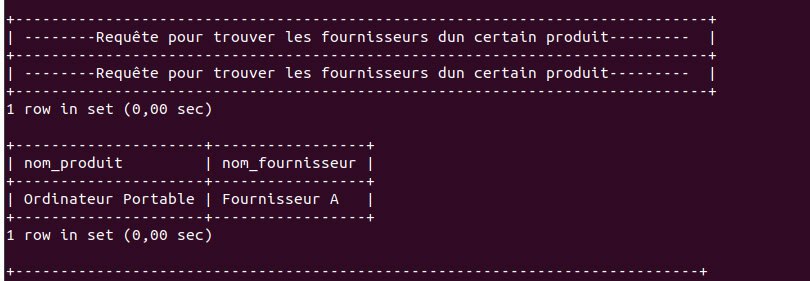
### création des requêtes



* affichage des résultats des requêtes

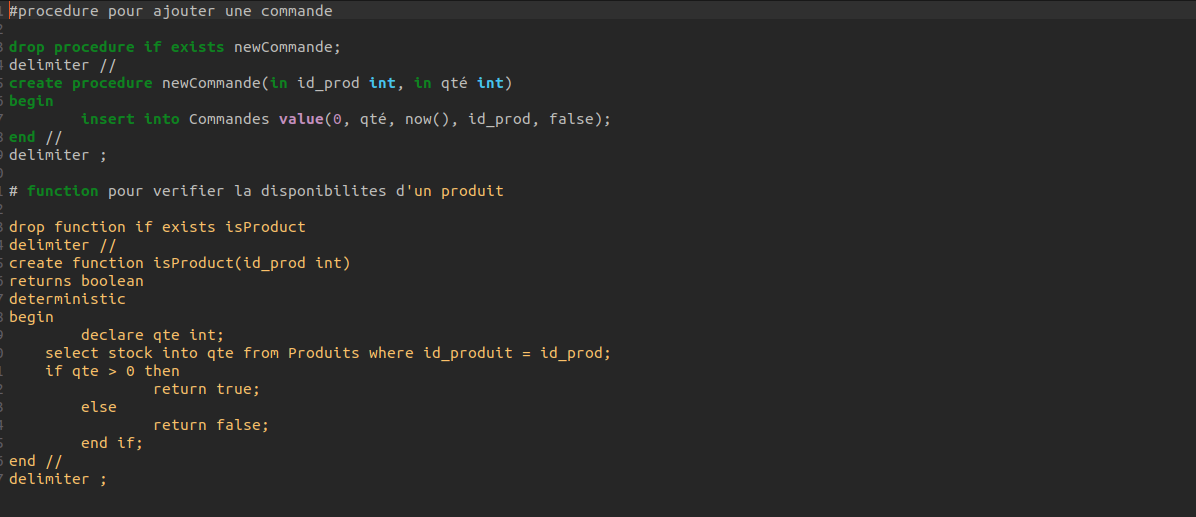




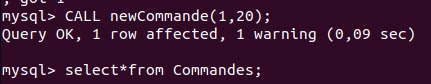


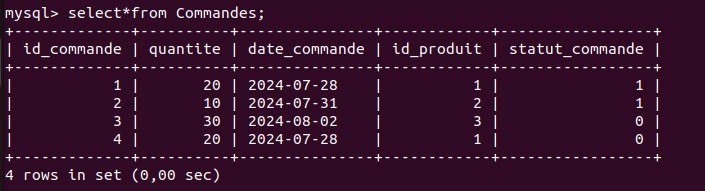


### Définition des fonction et procédures de notre système

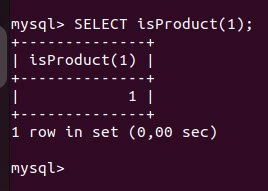


* résultats de la procédure pour créer une nouvelle commande(4)

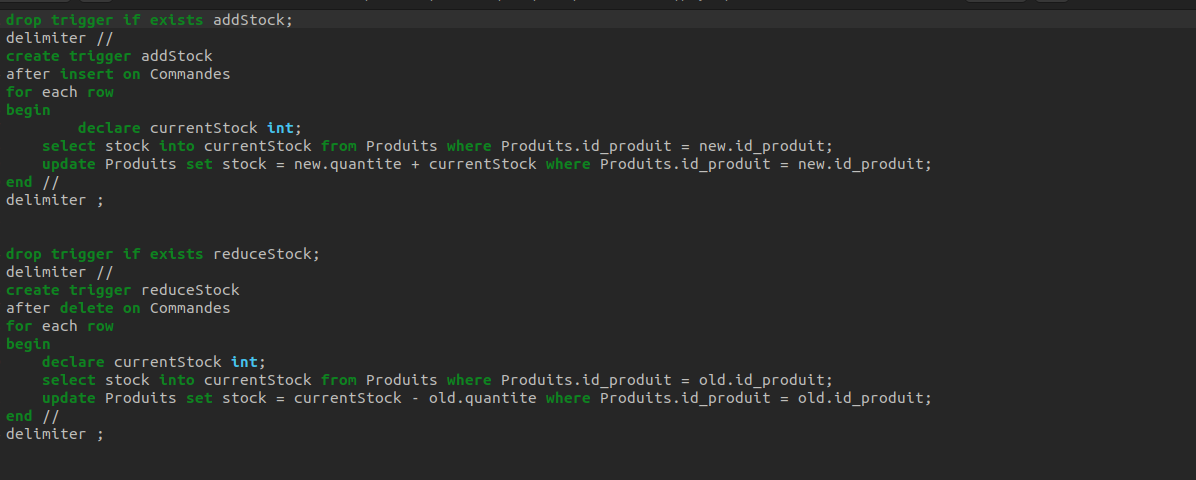


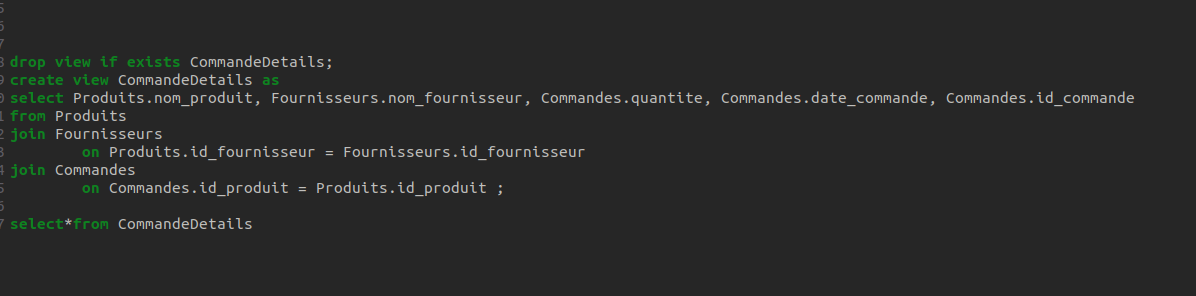


* résultats de la fonction pour vérifier la disponibilité d’un produit (isProduct)

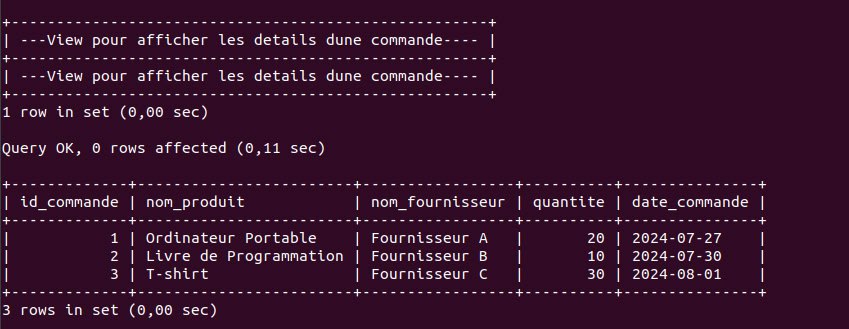


### mise en place des vues et triggers





* affichage des résultats de la vue



## Conclusion

A travers notre base de données que nous avons créé, nous avons simplifié la collecte des données et amélioré l'efficacité de la gestion des stocks du supermarché. Il n’est plus nécessaire d'utiliser des formulaires papier pour le suivi des produits et des commandes passées aux différents fournisseurs.